NeoVerse: Enhancing 4D World Model with in-the-wild Monocular Videos

Yuxue Yang[1,3] **Lue Fan**[1]✉ † **Ziqi Shi**[1] **Junran Peng**[2] **Feng Wang**[3] **Zhaoxiang Zhang**[1]✉

[1]NLPR & MAIS, Institute of Automation, Chinese Academy of Sciences
[2]University of Science and Technology Beijing [3]CreateAI

{yangyuxue2023, lue.fan}@ia.ac.cn
https://neoverse-4d.github.io

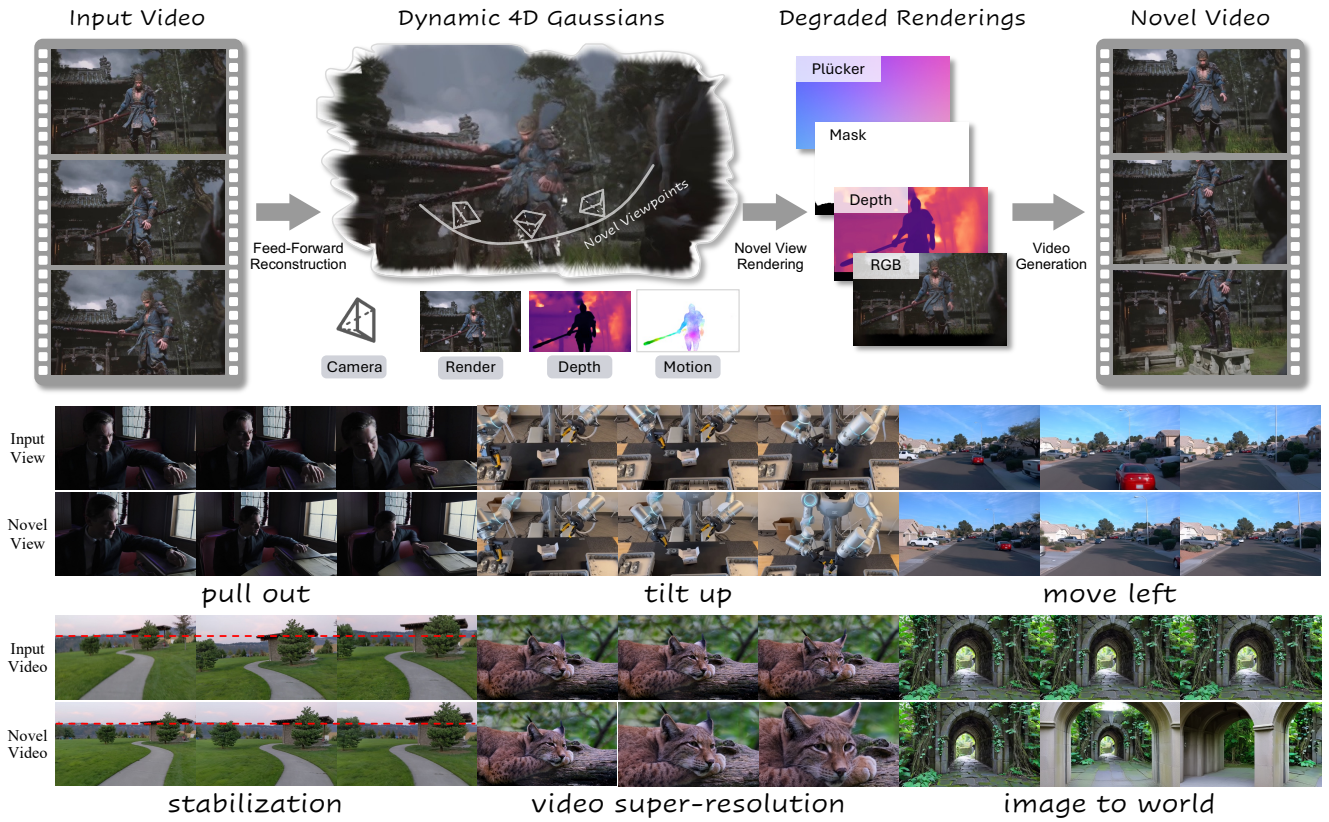


Figure 1. **Illustration of NeoVerse.** NeoVerse reconstructs 4D Gaussian Splatting (4DGS) from monocular videos in a feed-forward manner. These 4DGS can be rendered from novel viewpoints to provide degraded rendering conditions for generating high-quality and spatial-temporally coherent videos.

Abstract

*In this paper, we propose **NeoVerse**, a versatile 4D world model that is capable of 4D reconstruction, novel-trajectory video generation, and rich downstream applications. We first identify a common limitation of scalability in current 4D world modeling methods, caused either by expensive and specialized multi-view 4D data or by cumbersome training pre-processing. In contrast, our NeoVerse is built upon a core philosophy that makes the full pipeline scalable to diverse in-the-wild monocular videos. Specifically, NeoVerse features pose-free feed-forward 4D reconstruction, online monocular degradation pattern simulation, and other well-aligned techniques. These designs empower NeoVerse with versatility and generalization to various domains. Meanwhile, NeoVerse achieves state-of-the-art performance in standard reconstruction and generation benchmarks.*

✉ Corresponding Authors. † Project Lead.

1. Introduction

4D world modeling holds transformative potential in many fields, such as digital content creation, autonomous driving, and embodied intelligence. Recent approaches have made strides from both 3D side [9, 27, 44, 51, 63, 77] and 4D side [8, 11, 16, 22, 25, 40, 45, 53, 55, 60, 76] with a principle of *hybrid reconstruction and generation*. This paradigm typically involves two stages: reconstructing a 3D/4D representation [6, 23, 33, 68, 78] of the scene, and then, using the geometric prior to guide generation models [20, 35, 52, 56, 71]. Such a reconstruction-generation hybrid paradigm has widely recognized promising features, including spatiotemporal consistency and precise viewpoint control. However, the current solutions usually have limitations in terms of ***scalability***.

The limitation of scalability manifests in two main aspects. (1) ***Limited data scalability***. Some methods, such as ViewCrafter [77], utilize videos of static scenes to create multi-view training data and learn to generate videos in novel trajectories. Although effective, they cannot be extended to 4D scenes. Some other methods, such as ReCamMaster [2] and SynCamMaster [3], depend on specialized, hard-to-capture multi-view dynamic videos to learn novel trajectory generation. Such non-scalable data limits the model's generalization and versatility. (2) ***Limited training scalability***. Another line of work [8, 16, 25, 76] utilizes more flexible data types but usually necessitates a cumbersome offline pre-processing stage to create training data. For example, TrajectoryCrafter generates training data using a heavy video depth estimator [23] in an offline manner. Similarly, previous work FreeSim [16] pre-reconstructs the Gaussian field to prepare training input, which utilizes offline reconstruction [10, 12] and may even rely on extra 3D detection methods [15, 34, 69, 75]. Such an offline curation usually leads to significant computational burden, storage consumption, inflexible training scheme tuning, and even disables online augmentations. The two kinds of limitations erect a barrier to leveraging the cheap and diverse in-the-wild monocular videos, constraining the potential for building more powerful models.

To address these challenges, we propose NeoVerse. The core philosophy of NeoVerse is ***making the full pipeline scalable to diverse in-the-wild monocular videos***, enhancing generalization and versatility of 4D world models. To implement our vision, we first propose a feed-forward 4DGS model, built upon VGGT [57]. This model not only "Gaussianizes" VGGT but also features a bidirectional motion modeling mechanism, which is crucial for efficient online reconstruction (Sec. 3.2) and applications requiring time control. We then incorporate this feed-forward model into the generation training process. During each training iteration, it efficiently reconstructs 4D scenes using sparse key frames from monocular videos in an online manner. In addition, efficient online monocular degradation simulations, including Gaussian culling and average geometry filter, are proposed to simulate degraded rendering patterns in novel trajectories and offer conditions for generation. Combining them together makes the whole training process scalable to diverse in-the-wild monocular videos (up to 1M clips) in terms of both training efficiency and technical feasibility. We summarize our contributions as follows.

- We propose NeoVerse, a 4D world modeling approach, which is scalable to and enhanced by diverse in-the-wild monocular videos.
- NeoVerse is versatile, enabling many applications, including 4D reconstruction, multiview video generation, video editing, stabilization, super-resolution, etc.
- NeoVerse achieves state-of-the-art results in both reconstruction and generation tasks.
- We will make the source code publicly available to decentralize general 4D world models by leveraging cheap and diverse in-the-wild monocular videos.

2. Related Works

Feed-forward Gaussian Reconstruction. Recent stereo and 3D geometry foundation models [29, 32, 36, 37, 42, 57, 61, 64, 67, 72, 78] can estimate dense depth, point maps, and even camera parameters in a single forward pass, thereby driving a shift in Gaussian Splatting from per-scene optimization to generalizable feed-forward reconstruction. For static scenes, pose-free models such as NoPoSplat [72] reconstruct 3D Gaussians directly from sparse, unposed multi-view images, and AnySplat [29] further extends this paradigm to casually captured, long uncalibrated image sequences. For dynamic scenes, 4DGT [67], StreamSplat [64], and MoVieS [36] push feed-forward GS into 4D; however, each method still retains specific constraints: 4DGT is trained on posed monocular videos and adopts a largely uni-directional temporal modeling strategy, MoVieS similarly assumes known camera poses during training and inference, while StreamSplat focuses on frame-by-frame modeling.

Reconstruction-based Video Generation. Recent methods such as GEN3C [51], DaS [19], See3D [44], ViewCrafter [77], Difix3D+ [63], Voyager [27], Uni3C [8], FreeSim [16], TrajectoryCrafter [76], See4D [43], PostCam [11], Light-X [39] follow a hybrid *reconstruction+generation* paradigm, where a 3D/4D representation is first reconstructed and then used as geometric guidance for a generative video model. GEN3C [51] builds a depth-based 3D feature cache whose renderings condition a video diffusion model for 3D-consistent, pose-controllable synthesis; ViewCrafter [77] adopts a point-conditioned video diffusion framework to extend single-

or sparse-view inputs into long-range, high-fidelity novel-view sequences; Difix3D+ [63] applies a single-step diffusion enhancer to rendered novel views to correct artifacts in underconstrained regions and distill the improvements back into NeRF/3DGS representations; and TrajectoryCrafter [76] formulates camera-controllable video generation for monocular videos as trajectory redirection, conditioning a dual-stream diffusion backbone on point-cloud renderings and source frames to follow user-specified camera paths. Despite their strong spatial–temporal consistency and viewpoint controllability, these reconstruction-based approaches are mostly tailored to static or quasi-static scenes and rely on curated data or heavyweight offline reconstruction, limiting scalability to in-the-wild monocular videos.

3. Methodology

This section is organized as follows. In Sec. 3.1, we first propose an efficient pose-free feed-forward 4DGS reconstruction model, which reconstructs 4DGS from monocular videos. In Sec. 3.2, we introduce how to combine reconstruction part and generation and make the full pipeline scalable. Sec. 3.3 contains the training scheme and Sec. 3.4 elaborates on inference strategies.

3.1. Pose-Free Feed-Forward 4DGS Reconstruction

Our feed-forward model is partially built upon VGGT [57] backbone. For simplicity, we mainly introduce how we make VGGT dynamic and "Gaussianized".

Bidirectional motion modeling. Given a monocular video $\{\boldsymbol{I}_t \in \mathbb{R}^{H\times W\times 3}\}_{t=1}^{T}$, VGGT extracts the frame-wise features using the pretrained DINOv2 [47]. These features, concatenated with camera tokens and register tokens, are fed into a series of Alternating-Attention blocks [57], obtaining so-called ***frame features***. While this process effectively aggregates spatial information, they are insufficient for motion modeling due to temporal unawareness.

We introduce a bidirectional motion-encoding branch. Different from uni-directional motion in 4DGT [67], the bidirectional prediction distinguishes the instantaneous velocity between $t \to t+1$ and $t \to t-1$. Such a distinction facilitates temporal Gaussian interpolation between two consecutive timestamps.

Specifically, for the frame features $\{\boldsymbol{F}_t\}_{t=1}^{T}$, we copy and slice them into two parts along the temporal dimension: $\{\boldsymbol{F}_t\}_{t=1}^{T-1}$ and $\{\boldsymbol{F}_t\}_{t=2}^{T}$. Then we obtain **forward motion features** using the first part as queries and the second part as keys and values. Similarly, the **backward motion features** are encoded conversely. Formally, we have

$$\begin{aligned}\{\boldsymbol{F}_t^{\text{fwd}}\}_{t=1}^{T-1} &= \text{CrossAttn}(q=\{\boldsymbol{F}_t\}_{t=1}^{T-1}; k, v=\{\boldsymbol{F}_t\}_{t=2}^{T}),\\ \{\boldsymbol{F}_t^{\text{bwd}}\}_{t=2}^{T} &= \text{CrossAttn}(q=\{\boldsymbol{F}_t\}_{t=2}^{T}; k, v=\{\boldsymbol{F}_t\}_{t=1}^{T-1}),\end{aligned} \tag{1}$$

where $\boldsymbol{F}_t^{\text{fwd}}$ and $\boldsymbol{F}_t^{\text{bwd}}$ are forward motion features from timestamp t to $t+1$, and backward motion features from t to $t-1$. These features will be utilized to predict bidirectional linear and angular velocity of Gaussian primitives.

Gaussianizing VGGT. We first define 4D Gaussians as

$$\{(\boldsymbol{\mu}_i, \boldsymbol{\alpha}_i, \boldsymbol{r}_i, \boldsymbol{s}_i, \boldsymbol{sh}_i, \boldsymbol{\tau}_i, \boldsymbol{v}_i^{+}, \boldsymbol{v}_i^{-}, \boldsymbol{\omega}_i^{+}, \boldsymbol{\omega}_i^{-})\}_{i=1}^{T\times H\times W}, \tag{2}$$

where each Gaussian i is parameterized by: 3D position $\boldsymbol{\mu}_i$, opacity $\boldsymbol{\alpha}_i$, rotation $\boldsymbol{r}_i$, scale $\boldsymbol{s}_i$, and spherical harmonics coefficients $\boldsymbol{sh}_i$, as inherited from 3D Gaussians [33]. For bidirectional motion modeling, we introduce forward and backward velocities $\boldsymbol{v}_i^{+}$, $\boldsymbol{v}_i^{-}$, and forward and backward angular velocities $\boldsymbol{\omega}_i^{+}$, $\boldsymbol{\omega}_i^{-}$. In addition, we adopt a life span $\boldsymbol{\tau}_i$ following the common practice in 4DGS.

The 3D positions $\{\boldsymbol{\mu}_i\}$ is obtained by back-projecting pixel depth to 3D space using predicted depth and camera parameters. For the other attributes, $\{(\boldsymbol{\mu}_i, \boldsymbol{\alpha}_i, \boldsymbol{r}_i, \boldsymbol{s}_i, \boldsymbol{sh}_i, \boldsymbol{\tau}_i\}$ are predicted from the frame features, while the dynamic attributes $\{\boldsymbol{v}_i^{+}, \boldsymbol{v}_i^{-}, \boldsymbol{\omega}_i^{+}, \boldsymbol{\omega}_i^{-}\}$ are predicted from the bidirectional motion features.

3.2. Reconstruction-guided Video Generation

In this subsection, we introduce how to combine the reconstruction and generation in a scalable training pipeline.

Efficient on-the-fly reconstruction from sparse key frames. Although the proposed feed-forward 4DGS reconstruction is efficient, it can still be the bottleneck of training efficiency if we conduct on-the-fly reconstruction with long video input. To boost the training efficiency, we propose reconstruction from sparse key frames.

Given a long video input with N frames, we only take K key frames as reconstruction input but **render from all the N frames** since the rendering process is extremely efficient compared with network computation. However, such an operation requires interpolating the Gaussian field at non-keyframes. Thanks to our bidirectional motion modeling, such interpolation can be implemented as follows.

Given a non-key-frame query timestamp t_q, we transfer a nearest key-frame Gaussian i at timestamp t to t_q following

$$\boldsymbol{\mu}_i(t_q) = \begin{cases}\boldsymbol{\mu}_i + \boldsymbol{v}_i^{+}|t_q - t|, & t_q \geq t,\\ \boldsymbol{\mu}_i + \boldsymbol{v}_i^{-}|t_q - t|, & t_q < t,\end{cases} \tag{3}$$

$$\boldsymbol{r}_i(t_q) = \begin{cases}\boldsymbol{r}_i \cdot \phi(\boldsymbol{\omega}_i^{+}|t_q - t|), & t_q \geq t,\\ \boldsymbol{r}_i \cdot \phi(\boldsymbol{\omega}_i^{-}|t_q - t|), & t_q < t,\end{cases} \tag{4}$$

$$\boldsymbol{\alpha}_i(t_q) = \boldsymbol{\alpha}_i \exp(-\gamma \cdot d(t_q, t)^{\frac{1}{1-\boldsymbol{\tau}_i}}), \tag{5}$$

where we assume the real-world motion in a short interval between two adjacent input frames is approximately linear.

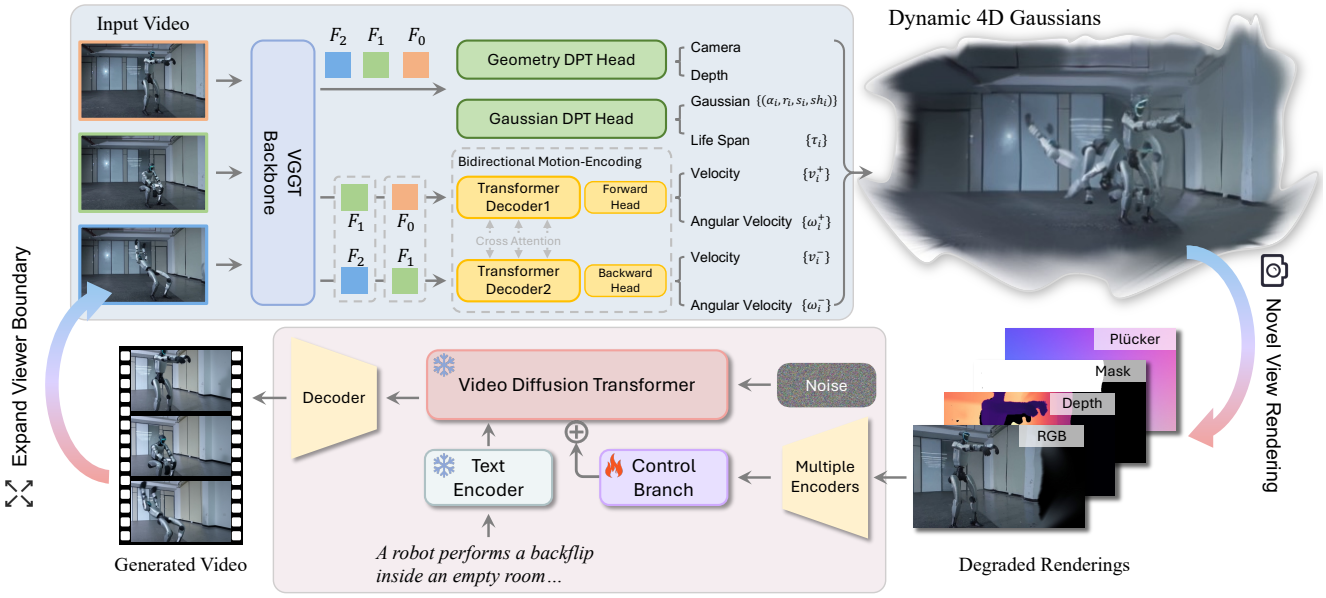
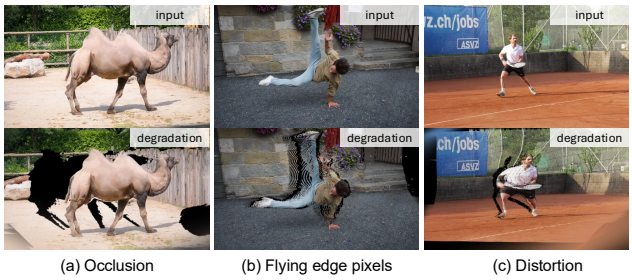


Figure 2. **Framework of NeoVerse.** In the reconstruction part, we propose a pose-free feed-forward 4DGS reconstruction model (Sec. 3.1) with bidirectional motion modeling. The degraded renderings in novel viewpoints from 4DGS are input to the generation model as conditions. During training, the degraded rendering conditions are simulated from monocular videos (Sec. 3.2), and the original videos themselves serve as targets.



(a) Occlusion (b) Flying edge pixels (c) Distortion

Figure 3. **Training pairs with degradation simulation.**

Angular velocities $\boldsymbol{\omega}_i^{\pm}$ are represented in the axis-angle representation, and $\phi(\cdot)$ converts it to a quaternion. The opacity of the Gaussian is represented by a time-varying function to ensure a natural transition between input frames. To handle non-uniform keyframe intervals, we model opacity decay with a normalized temporal distance $d(t_q, t) = \frac{|t_q - t|}{|T_{k+1} - T_k|} \leq 1$, where $[T_k, T_{k+1}]$ is the keyframe interval containing query timestamp t_q. The life span $\boldsymbol{\tau}_i$ is constrained in the range of $(0, 1)$ with a sigmoid function, and γ is a hyper-parameter that controls the decay speed. When $\boldsymbol{\tau}_i$ approaches 1, the $\exp(\cdot)$ tends towards 1, indicating $\boldsymbol{\alpha}_i(t_q) \approx \boldsymbol{\alpha}_i$; otherwise, $\boldsymbol{\alpha}_i(t_q)$ decays rapidly.

Monocular degradation simulation. Our generation model is expected to generate high-quality novel views from low-quality novel view renderings, necessitating such training pairs. For multi-view or static datasets [38, 74], we can easily get such training pairs as in ViewCrafter [77]. However, for in-the-wild monocular videos, we need to carefully simulate degradation renderings paired with ground-truth monocular frames. Therefore, we propose three techniques to simulate the degradation rendering patterns based on monocular videos.

(1) ***Visibility-based Gaussian Culling*** *for occlusion simulation.* Given the camera pose trajectory predicted from the sparse key frames, we apply a random transform to the trajectory to obtain a novel trajectory. A constraint is applied to this transform to ensure new camera poses still roughly point to the scene center. Using depth, we can easily identify those Gaussians that are occluded from the transformed new camera poses. We then simply cull those invisible Gaussian primitives and render the remaining Gaussian primitives back into the original viewpoints, resulting degradation pattern demonstrated in Fig. 3 (a).

(2) ***Average Geometry Filter*** *for flying-edge-pixel and distortion simulation.* In addition to the occlusion, another typical degradation pattern is the flying pixels in depth-discontinuous edges. The network has tendency to produce ***average*** depth value at those edges to minimize regression loss, as also confirmed by [65]. From a first-principles perspective, we propose to use a ***average filter*** to create such averaged depth patterns. Specifically, we render depth in the transformed novel trajectory and apply an average filter in the rendered depth map. We then adjust the center position of each Gaussian according to the average filtered depth value. When such modified Gaussians are rendered back into the original views, the flying-pixel pattern appears as shown in Fig. 3 (b). We can further apply a larger filter kernel to simulate spatially broader distortions shown in Fig. 3

(c), caused by potential depth error.

All three kinds of degradations in Fig. 3 are simulated with fundamental principles in geometry relation and depth learning, and designed to be simple yet effective, enabling the utilization of in-the-wild monocular videos.

Degraded rendering conditioning. We use the obtained degraded renderings as conditions for generation and the original videos as targets. The rendered conditions include multiple modalities, including RGB images, depth maps, and masks binarized from opacity maps to indicate the empty regions. Plüker embeddings of the original trajectory are also computed to provide explicit 3D camera motion information [8]. We introduce a control branch to incorporate them into the generation model like [27, 30, 70, 79]. During training, we only train the control branch while freezing the video generation model, not only for training efficiency, but more importantly, to make NeoVerse accessible to powerful distillation LoRAs [21] to speed up the generation process.

3.3. Training Scheme

We partition the training into two stages: 1) reconstruction model training; 2) generation model training with on-the-fly reconstruction and degradation simulation.

Reconstruction. We train our feed-forward 4DGS reconstruction model with a multi-task loss on various static and dynamic 3D datasets:

$$\mathcal{L}_{\text{recon}} = \mathcal{L}_{\text{rgb}} + \lambda_1 \mathcal{L}_{\text{camera}} + \lambda_2 \mathcal{L}_{\text{depth}} + \lambda_3 \mathcal{L}_{\text{motion}} + \lambda_4 \mathcal{L}_{\text{regular}}, \tag{6}$$

where $\mathcal{L}_{\text{rgb}}$ is the photometric loss between rendered and ground-truth images, including an L_2 loss and LPIPS [80] loss. The camera loss $\mathcal{L}_{\text{camera}}$ and depth loss $\mathcal{L}_{\text{depth}}$ supervise the predicted camera parameters and depth maps following VGGT [57]. Notably, $\mathcal{L}_{\text{depth}}$ also contains the supervision for rendered depth from Gaussians. The motion loss $\mathcal{L}_{\text{motion}} = \sum_i \|\hat{\boldsymbol{v}}_i^+ - \boldsymbol{v}_i^+\| + \|\hat{\boldsymbol{v}}_i^- - \boldsymbol{v}_i^-\|$ adds supervision on the predicted bidirectional velocities, where $\hat{\boldsymbol{v}}_i^+$ and $\hat{\boldsymbol{v}}_i^-$ are the ground-truth forward and backward velocities computed from some dynamic 3D datasets [7, 18, 31, 46, 54, 82]. To prevent the Gaussians from becoming erroneously transparent, we introduce a regularization loss $\mathcal{L}_{\text{regular}} = \sum_i |1 - \boldsymbol{A}_i|$, where $\boldsymbol{A}_i$ is rendered accumulated opacity map.

Generation. For generation model training, we adopt Rectified Flow [14] and Wan-T2V [56] 14B to model the denoising diffusion process. **The whole training process is performed on monocular videos.** Given a monocular video, we first utilize on-the-fly reconstruction from sparse key frames to obtain 4DGS and simulate degradation renderings as conditions c_{render}. For the video latent x_1 and sampled noise $x_0 \sim \mathcal{N}(0, I)$, the training objective of generation model f_θ is formulated as

$$\mathcal{L}_{\text{gen}} = \mathbb{E}_{x_1, x_0, c_{\text{render}}, c_{\text{text}}, t} \| f_\theta(x_t, t, c_{\text{render}}, c_{\text{text}}) - v_t \|_2^2, \tag{7}$$

where x_t is a linear interpolation between x_1 and x_0 at timestamp t, $v_t = x_1 - x_0$ is ground-truth velocity. c_{text} is the text condition extracted from the video caption using a language model like umT5 [13]. Renderings c_{render} are input into the generation model through a control branch like [30, 79].

3.4. Inference

Reconstruction and global motion tracking. Given a monocular video, our feed-forward model outputs 4DGS and camera parameters of each frame. Before rendering conditions from a novel trajectory, we can optionally aggregate Gaussians from multiple timestamps into a single timestamp for a more complete representation. For better aggregation, we conduct motion separation by global motion tracking.

The motivation of global motion tracking is to identify those objects undergoing both static and dynamic phases in a clip, which should be regarded as the dynamic part and cannot be easily identified using predicted instantaneous velocity. Taking a Gaussian primitive i as example, given world-to-camera poses $\{\boldsymbol{P}_t\}_{t=1}^T$, camera intrinsics $\{\boldsymbol{K}_t\}_{t=1}^T$, and Gaussian position $\boldsymbol{\mu}_i$ for Gaussian i, we project the Gaussian center to each frame t and compute its projected pixel coordinates $\boldsymbol{p}_{i,t}$ and depth $\boldsymbol{d}_{i,t}$. Let $D_t[\boldsymbol{p}_{i,t}]$ and $V_t[\boldsymbol{p}_{i,t}]$ are the sampled depth and velocity at pixel $\boldsymbol{p}_{i,t}$. We define a visibility-weighted maximum velocity magnitude at the global video level as

$$\begin{aligned} \boldsymbol{m}_{i,t} &= \max\{\|V_t^+[\boldsymbol{p}_{i,t}]\|_2, \|V_t^-[\boldsymbol{p}_{i,t}]\|_2\}, \\ \boldsymbol{m}_i &= \max_{t=1,\dots,T} \mathbb{1}(\boldsymbol{d}_{i,t} \le D_t[\boldsymbol{p}_{i,t}]) \cdot \boldsymbol{m}_{i,t}, \end{aligned} \tag{8}$$

where $\boldsymbol{m}_{i,t}$ is the maximum velocity magnitude at frame t, $\mathbb{1}(\cdot)$ is a function indicating whether the Gaussian is visible, and $\boldsymbol{m}_i$ is the visibility-weighted maximum velocity magnitude across all frames. Finally, we separate the Gaussians into static set $\mathcal{S}$ and dynamic set $\mathcal{D}$ according to $\boldsymbol{m}_i$ with a threshold η.

Temporal aggregation, interpolation, and generation. With a separated dynamic part and a static part, we conduct two different Gaussian temporal aggregation strategies for each part, respectively. The static part is simply aggregated across all frames, while the dynamic part is aggregated only from a couple of nearby frames to avoid motion drifting errors.

In some cases, we may need to interpolate Gaussians into an intermediate timestamp between two adjacent discrete frames. A typical case is creating slow-motion videos

and bullet-time shots. Our bidirectional motion mechanism sufficiently supports such tasks happening in a short time interval. In practice, we use similar techniques in Sec. 3.2 for interpolation.

After the optional aggregation and interpolation, we render the resulting Gaussians into any desired novel trajectory. The renderings, along with other conditions, are sent to the generation model to generate videos.

4. Experiments

4.1. Implementation

For reconstruction, we follow the learning rate schedule of VGGT [57]. We resize all input videos to have a longest edge of 560 pixels. GSplat [73] is adopted as the Gaussian Splatting rendering backend. For the generation, the video resolution is fixed at 336×560 and the length is set to 81 frames. The training is conducted on 32 A800 GPUs, where the first stage trains 150K iterations and the second stage trains 50K iterations. More training details can be found in the supplementary material.

Datasets. We collect 18 public datasets following CUT3R [59], including Arkitscenes [4], DL3DV [38], PointOdyssey [82], Kubric [18], Waymo [54], SpatialVID [58], GFIE [24], etc. Besides the above datasets, we further curate a large-scale self-collected monocular video dataset from the internet, containing over 1M videos from diverse scenarios. More details about datasets are provided in the supplementary material.

4.2. Quantitative Evaluation

Reconstruction benchmark. Our reconstruction results on both static and dynamic datasets are shown in Table 1 and Table 2, respectively. Our reconstruction part achieves state-of-the-art performance among almost all metrics. Recent reprints MoVieS [36] and StreamSplat [64] are not listed in the table because they are neither open-sourced nor provide a detailed evaluation protocol. Our detailed evaluation protocols are provided in the supplementary material.

Method	VRNeRF [66] (16 views)			Scannet++ [74] (32 views)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NoPoSplat [72]	11.27	0.408	0.620	8.69	0.312	0.614
Flare [81]	12.62	0.597	0.623	12.19	0.619	0.611
AnySplat [29]	18.02	0.705	0.366	22.79	0.773	0.217
Ours	**20.73**	**0.766**	**0.352**	**25.34**	**0.834**	**0.195**

Table 1. Quantitative comparison with other **static** reconstruction models.

Generation benchmark. In Table 3, we compared the generation performance with related work TrajectoryCrafter [76] and ReCamMaster [2], demonstrating better performance. We conduct more analysis in the section of qualitative evaluation.

Method	ADT [48]			DyCheck [17]		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
MonST3R [78]	17.42	0.554	0.534	9.32	0.103	0.710
4DGT† [67]	30.09	0.909	0.178	9.94	0.208	0.639
Ours	**32.56**	**0.927**	**0.120**	**11.56**	**0.293**	**0.558**

Table 2. Quantitative comparison with other **dynamic** reconstruction models. †: indicate the method takes camera poses as input.

Runtime evaluation. Table 3 also shows the efficiency evaluation of both the reconstruction stage and the generation stage. Thanks to our intentional design of condition injection in Sec. 3.2, our generation process gets significantly accelerated by the off-the-shelf distillation technique. More importantly, as discussed in Sec. 3.2, our bidirectional motion design enables more efficient reconstruction from sparse key frames without loss of generation performance.

4.3. Qualitative Evaluation and Analysis

For an intuitive understanding, we conduct rich qualitative evaluations and analysis, leading to the following findings.

Rendering quality. Fig. 5 and Fig. 6 demonstrate the rendering quality comparison. Our model not only achieves better visual quality but is also more faithful to input observations. Instead, other methods may predict unreal artifacts such as regions indicated by yellow boxes in Fig. 5.

Pose prediction accuracy. It is noteworthy that our model also has better pose prediction accuracy. In Fig. 5, the compared method [29] shows a field of view (images with red boundaries) inconsistent with the ground truth, which is caused by inaccurate pose prediction.

Trajectory controlability vs. generation quality. An intriguing and fundamental phenomenon we can find in Fig. 4 is that related work usually demonstrates a trade-off between generation quality and trajectory controllability. Specifically, TrajectoryCrater, a reconstruction-generation hybrid method similar to our NeoVerse , shows good trajectory controllability and exhibits consistent trajectories with our method, while its generation quality is inferior. This is mainly caused by its non-scalable training pipeline, stopping the model from seeing diverse in-the-wild videos, such as very challenging human activities in Fig. 4.

In contrast, the purely generation-based method ReCamMaster shows good visual generation quality, but cannot achieve precise trajectory control, which is crucial in some downstream tasks such as simulation.

Artifact suppression. Another reason for our superiority over the similar reconstruction-based TrajectoryCrafter is that our degradation simulations (Fig. 3) enable artifact

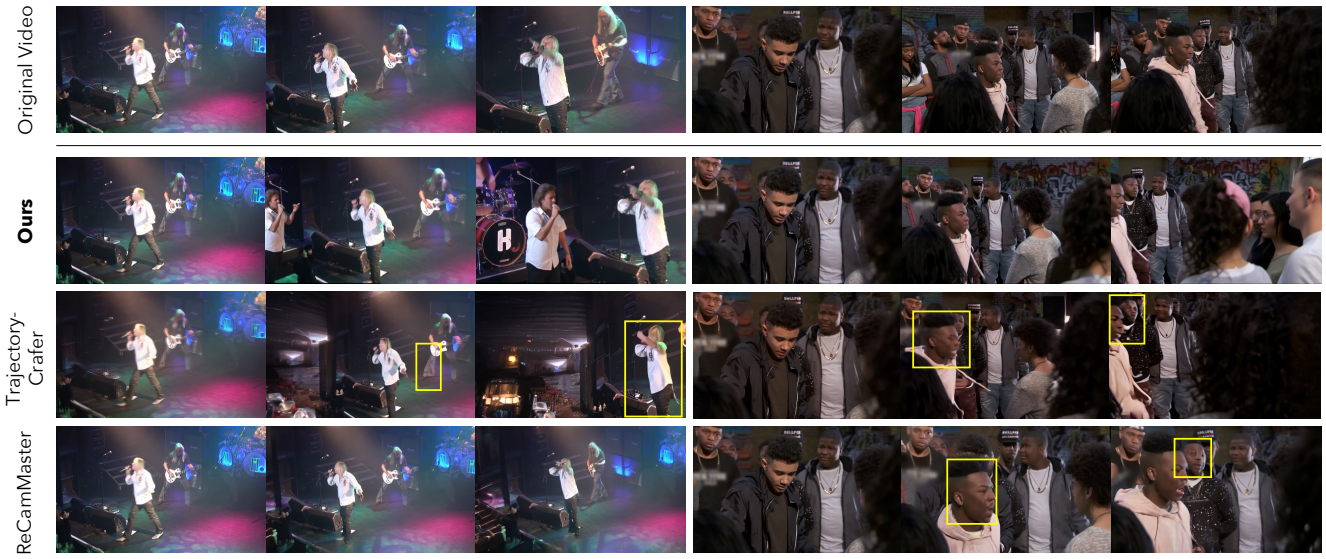


Figure 4. **Generation with large camera motions on challenging in-the-wild videos.** We compare our method against other related work on "Pan left" (left) and "Move right" (right) cases. Our NeoVerse achieves better generation quality while maintaining precise camera controllability. Yellow boxes highlight artifacts.

Method	Frames	Inference Time (s) Recon.	Gen.	Total	Subj. Consist.	Back. Consist.	Temp. Flick.	Motion Smooth.	Aesth. Quality	Imag. Quality
TrajectoryCrafter [76]	49	38	121	159	83.02	88.58	94.71	97.64	44.63	54.59
ReCamMaster [2]	81	-	168	168	88.21	91.60	96.56	**98.86**	44.29	58.87
Ours (11 key frames)	81	2	18	**20**	88.43	92.27	**96.77**	98.80	44.55	59.75
Ours (21 key frames)	81	3	18	21	88.73	92.43	96.76	98.71	44.59	60.01
Ours (41 key frames)	81	5	18	23	89.10	92.65	96.67	98.63	**44.89**	60.37
Ours (full frames)	81	10	18	28	**89.42**	**92.79**	96.51	98.67	44.78	**61.51**

Table 3. **VBench [28] results for novel view generation.** We randomly collect 100 unseen in-the-wild videos, each with 4 different camera trajectories, resulting in a total of 400 test cases. For a fair comparison of inference time, we resize all videos to 336×560 resolution and report the average results over all test cases. The runtime evaluation is conducted on an A800 GPU.

suppression. In contrast, the generation quality of TrajectoryCrafter is significantly decreased by "ghosting patterns" from inaccurate reconstruction.

Contextually grounded imagination. Fig. 4 also demonstrates that our NeoVerse can conduct contextually grounded imagination for non-observed regions, such as the second singer and crowded people. We give credit to our design scalability to diverse in-the-wild videos.

4.4. Ablation Study

Method	PSNR↑	SSIM↑	LPIPS↓
w/o Regularization	10.86	0.244	0.576
w/o Bidirectional Motion	11.27	0.285	0.570
Reconstruction part	11.56	0.293	0.558
w/ Generation	14.59	0.323	0.501

Table 4. **Ablation experiments on DyCheck.** "w/. Generation" indicates our full pipeline, which gains significant performance improvements over the pure reconstruction part.

Motion modeling. In Table 4, we remove the motion modeling mechanism by skipping Eq. (1) and predicting motions directly from frame features. The performance drop reveals the effectiveness of our modeling mechanism.

Opacity regularization. In Sec. 3.3, we introduce opacity regularization to avoid the model learning a shortcut, which is outputting transparent primitives for the regions in similar colors to the predefined background color. This technique is proven effective in Table 4.

Degradation simulation. As discussed in Sec. 3.2, large camera motions often result in degraded renderings containing flying edge pixels and distortions. Fig. 7 demonstrates the necessity of our online degradation simulation. Without training on simulated degraded samples, the generation model tends to trust the geometric artifacts in the condition, leading to "ghosting" effects or blurred outputs. **By incorporating degradation simulation, the model learns to suppress these artifacts and hallucinate realistic details in occluded or distorted regions**.

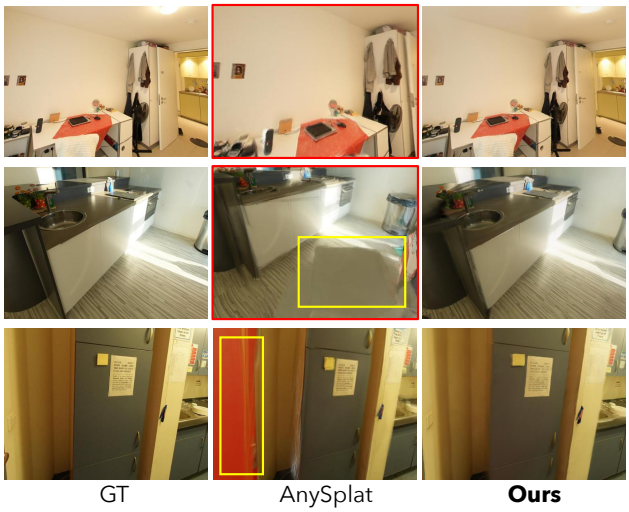


Figure 5. Qualitative comparison with state-of-the-art methods in **static scenes**. Red boundaries indicate inconsistent renderings due to inaccurate pose prediction. Yellow boxes indicate artifacts.

Figure 6. Qualitative comparison with state-of-the-art methods in **dynamic scenes**. Yellow boxes indicate artifacts. Note that the black regions in our prediction are *not error* but mainly caused by partial observations of input frames.

Global motion tracking. Fig. 8 showcases the importance of global motion tracking when identifying the dynamic instances. Without the global tracking, some dynamic objects are mistakenly identified as static due to a partial static state.

4.5. Applications

A superiority of NeoVerse is the support for rich downstream applications other than the novel trajectory video generation. Due to the limited space, here we briefly introduce several typical applications, leaving more details in the supplementary materials.

3D tracking. By associating nearest Gaussian primitives between consecutive frames using predicted 3D flow, our NeoVerse achieves 3D tracking shown in Fig. 9.

Figure 7. **Effectiveness of degradation simulation.** The model learns to suppress artifacts and hallucinate realistic details in occluded or distorted regions through degradation simulation.

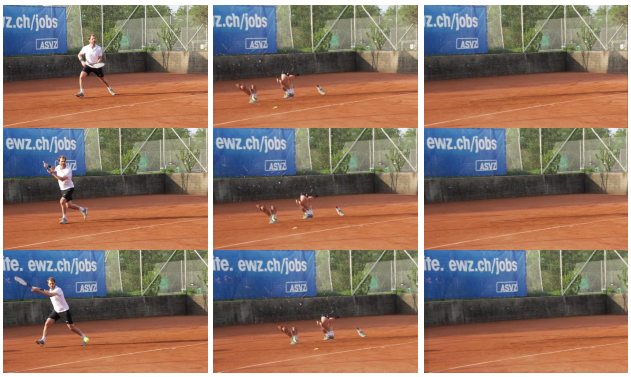


Figure 8. **Visualization about global motion tracking and aggregation**. (a) Input video. (b) Aggregated static Gaussians separated by predicted velocities. (c) Aggregated static Gaussians separated with global motion tracking.

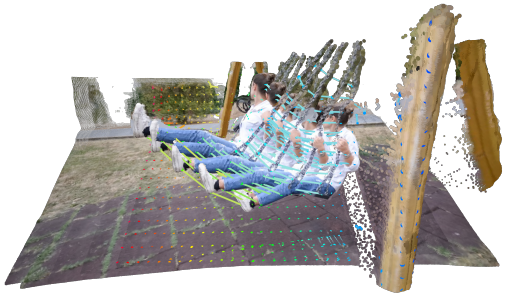


Figure 9. **Visualization of 3D tracking.** For better visualization, we only show the Gaussian centers.

Video editing. Since our model has a binary mask condition and a textual condition, it can edit videos with the help of a video segmentation model [50], demonstrated in Fig. 10.

Figure 10. **Video editing.** Left: The white car is edited to be red. Right: The mirror teapot is edited to be transparent.

Video stabilization. By smoothing the predicted camera trajectory, our model achieves effective video stabilization, as demonstrated in the teaser Fig. 1.

Video super-resolution The Gaussian representation in NeoVerse supports flexible rendering resolution without the significant loss of appearance information. Thus, NeoVerse can achieve video super-resolution by generation with a larger rendering resolution, also demonstrated in Fig. 1.

Others. Moreover, NeoVerse is also capable of other applications such as background extraction (Fig. 8), image to world (Fig. 1). We leave more demonstrations in the supplementary materials.

5. Conclusion and Limitations

In this paper, we introduce **NeoVerse**, a 4D world model that overcomes key scalability limitations in previous arts, building a training pipeline scalable to in-the-wild monocular videos. Thus, the generalization and versatility of NeoVerse are significantly enhanced by the diverse in-the-wild data, enabling various downstream applications. Extensive experiments demonstrate state-of-the-art performance in both reconstruction and generation tasks.

Limitations. NeoVerse requires data with correct underlying 3D information. Therefore, it cannot be trivially applied to data without 3D information like 2D cartoons. Due to the constraints of training resources, our curated dataset (1M clips) is not that large. We leave more data for future work.

References

[1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *ECCV*, pages 690–708. Springer, 2022. 1

[2] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. In *ICCV*, 2025. 2, 6, 7

[3] Jianhong Bai, Menghan Xia, Xintao Wang, Ziyang Yuan, Xiao Fu, Zuozhu Liu, Haoji Hu, Pengfei Wan, and Di Zhang. Syncammaster: Synchronizing multi-camera video generation from diverse viewpoints. In *ICLR*, 2025. 2

[4] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *arXiv preprint arXiv:2111.08897*, 2021. 6, 1

[5] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *CVPR*, pages 8726–8737, 2023. 1

[6] Aleksei Bochkovskii, AmaÃĢl Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 2

[7] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 5, 1

[8] Chenjie Cao, Jingkai Zhou, Shikai Li, Jingyun Liang, Chaohui Yu, Fan Wang, Xiangyang Xue, and Yanwei Fu. Uni3c: Unifying precisely 3d-enhanced camera and human motion controls for video generation. *arXiv preprint arXiv:2504.14899*, 2025. 2, 5

[9] Luxi Chen, Zihan Zhou, Min Zhao, Yikai Wang, Ge Zhang, Wenhao Huang, Hao Sun, Ji-Rong Wen, and Chongxuan Li. Flexworld: Progressively expanding 3d scenes for flexible-view synthesis. *arXiv preprint arXiv:2503.13265*, 2025. 2

[10] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *arXiv preprint arXiv:2311.18561*, 2023. 2

[11] Yipeng Chen, Zhichao Ye, Zhenzhou Fang, Xinyu Chen, Xiaoyu Zhang, Jialing Liu, Nan Wang, Haomin Liu, and Guofeng Zhang. Postcam: Camera-controllable novel-view video generation with query-shared cross-attention. *arXiv preprint arXiv:2511.17185*, 2025. 2

[12] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. *arXiv preprint arXiv:2408.16760*, 2024. 2

[13] Hyung Won Chung, Noah Constant, Xavier Garcia, Adam Roberts, Yi Tay, Sharan Narang, and Orhan Firat. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. *arXiv preprint arXiv:2304.09151*, 2023. 5

[14] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 5

[15] Lue Fan, Ziqi Pang, Tianyuan Zhang, Yu-Xiong Wang, Hang Zhao, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Embracing Single Stride 3D Object Detector with Sparse Transformer. In *CVPR*, 2022. 2

[16] Lue Fan, Hao Zhang, Qitai Wang, Hongsheng Li, and Zhaoxiang Zhang. Freesim: Toward free-viewpoint camera simula-

tion in driving scenes. In *CVPR*, pages 12004–12014, 2025. 2

[17] Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Monocular dynamic view synthesis: A reality check. *NIPS*, 35:33768–33780, 2022. 6, 2

[18] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *CVPR*, pages 3749–3761, 2022. 5, 6, 1

[19] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, et al. Diffusion as shader: 3d-aware video diffusion for versatile video generation control. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–12, 2025. 2

[20] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 5

[22] Tao Hu, Haoyang Peng, Xiao Liu, and Yuewen Ma. Ex-4d: Extreme viewpoint 4d video synthesis via depth watertight mesh. *arXiv preprint arXiv:2506.05554*, 2025. 2

[23] Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. Depthcrafter: Generating consistent long depth sequences for open-world videos. In *CVPR*, pages 2005–2015, 2025. 2

[24] Zhengxi Hu, Yuxue Yang, Xiaolin Zhai, Dingye Yang, Bohan Zhou, and Jingtai Liu. Gfie: A dataset and baseline for gaze-following from 2d to 3d in indoor environments. In *CVPR*, pages 8907–8916, 2023. 6, 1

[25] Jiaxin Huang, Sheng Miao, Bangbang Yang, Yuewen Ma, and Yiyi Liao. Vivid4d: Improving 4d reconstruction from monocular video by video inpainting. In *ICCV*, pages 12592–12604, 2025. 2

[26] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *CVPR*, pages 2821–2830, 2018. 1

[27] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson WH Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *arXiv preprint arXiv:2506.04225*, 2025. 2, 5

[28] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *CVPR*, pages 21807–21818, 2024. 7

[29] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *arXiv preprint arXiv:2505.23716*, 2025. 2, 6

[30] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025. 5

[31] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *CVPR*, pages 13229–13239, 2023. 5, 1

[32] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. *arXiv preprint arXiv:2509.13414*, 2025. 2

[33] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *TOG*, 42(4):139–1, 2023. 2, 3

[34] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[35] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, et al. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024. 2

[36] Chenguo Lin, Yuchen Lin, Panwang Pan, Yifan Yu, Honglei Yan, Katerina Fragkiadaki, and Yadong Mu. Movies: Motion-aware 4d dynamic view synthesis in one second. *arXiv preprint arXiv:2507.10065*, 2025. 2, 6

[37] Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025. 2

[38] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, pages 22160–22169, 2024. 4, 6, 1

[39] Tianqi Liu, Zhaoxi Chen, Zihao Huang, Shaocong Xu, Saining Zhang, Chongjie Ye, Bohan Li, Zhiguo Cao, Wei Li, Hao Zhao, et al. Light-x: Generative 4d video rendering with camera and illumination control. *arXiv preprint arXiv:2512.05115*, 2025. 2

[40] Tianqi Liu, Zihao Huang, Zhaoxi Chen, Guangcong Wang, Shoukang Hu, Liao Shen, Huiqiang Sun, Zhiguo Cao, Wei Li, and Ziwei Liu. Free4d: Tuning-free 4d scene generation with spatial-temporal consistency. *arXiv preprint arXiv:2503.20785*, 2025. 2

[41] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *CVPR*, pages 21013–21022, 2022. 1

[42] Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025. 2

[43] Dongyue Lu, Ao Liang, Tianxin Huang, Xiao Fu, Yuyang Zhao, Baorui Ma, Liang Pan, Wei Yin, Lingdong Kong, Wei Tsang Ooi, et al. See4d: Pose-free 4d generation via auto-regressive video inpainting. *arXiv preprint arXiv:2510.26796*, 2025. 2

[44] Baorui Ma, Huachen Gao, Haoge Deng, Zhengxiong Luo, Tiejun Huang, Lulu Tang, and Xinlong Wang. You see it, you got it: Learning 3d creation on pose-free videos at scale. In *CVPR*, pages 2016–2029, 2025. 2

[45] Yue Ma, Kunyu Feng, Xinhua Zhang, Hongyu Liu, David Junhao Zhang, Jinbo Xing, Yinhan Zhang, Ayden Yang, Zeyu Wang, and Qifeng Chen. Follow-your-creation: Empowering 4d creation through video inpainting. *arXiv preprint arXiv:2506.04590*, 2025. 2

[46] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *CVPR*, pages 4981–4991, 2023. 5, 1

[47] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3, 1

[48] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *ICCV*, pages 20133–20143, 2023. 6, 2

[49] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, pages 12179–12188, 2021. 1

[50] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 8

[51] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *CVPR*, pages 6121–6132, 2025. 2

[52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 2

[53] Chenxi Song, Yanming Yang, Tong Zhao, Ruibo Li, and Chi Zhang. Worldforge: Unlocking emergent 3d/4d generation in video diffusion model via training-free guidance. *arXiv preprint arXiv:2509.15130*, 2025. 2

[54] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, pages 2446–2454, 2020. 5, 6, 1

[55] Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *European Conference on Computer Vision*, pages 313–331. Springer, 2024. 2

[56] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2, 5, 1

[57] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, pages 5294–5306, 2025. 2, 3, 5, 6

[58] Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Yi Zhang, Chang Zeng, Yanxi Zhou, et al. Spatialvid: A large-scale video dataset with spatial annotations. *arXiv preprint arXiv:2509.09676*, 2025. 6, 1

[59] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, pages 10510–10522, 2025. 6

[60] Qisen Wang, Yifan Zhao, Peisen Shen, Jialu Li, and Jia Li. Chronosobserver: Taming 4d world with hyperspace diffusion sampling. *arXiv preprint arXiv:2512.01481*, 2025. 2

[61] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, pages 20697–20709, 2024. 2, 1

[62] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *IROS*, pages 4909–4916, 2020. 1

[63] Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *CVPR*, pages 26024–26035, 2025. 2, 3

[64] Zike Wu, Qi Yan, Xuanyu Yi, Lele Wang, and Renjie Liao. Streamsplat: Towards online dynamic 3d reconstruction from uncalibrated video streams. *arXiv preprint arXiv:2506.08862*, 2025. 2, 6

[65] Gangwei Xu, Haotong Lin, Hongcheng Luo, Xianqi Wang, Jingfeng Yao, Lianghui Zhu, Yuechuan Pu, Cheng Chi, Haiyang Sun, Bing Wang, et al. Pixel-perfect depth with semantics-prompted diffusion transformers. *arXiv preprint arXiv:2510.07316*, 2025. 4

[66] Linning Xu, Vasu Agrawal, William Laney, Tony Garcia, Aayush Bansal, Changil Kim, Samuel Rota Bulò, Lorenzo Porzi, Peter Kontschieder, Aljaž Božič, et al. Vr-nerf: High-fidelity virtualized walkable spaces. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–12, 2023. 6, 2

[67] Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4dgt: Learning a 4d gaussian transformer using real-world monocular videos. *arXiv preprint arXiv:2506.08015*, 2025. 2, 3, 6

[68] Honghui Yang, Di Huang, Wei Yin, Chunhua Shen, Haifeng Liu, Xiaofei He, Binbin Lin, Wanli Ouyang, and Tong He.

Depth any video with scalable synthetic data. *arXiv preprint arXiv:2410.10815*, 2024. 2

[69] Yuxue Yang, Lue Fan, and Zhaoxiang Zhang. Mixsup: Mixed-grained supervision for label-efficient lidar-based 3d object detection. *arXiv preprint arXiv:2401.16305*, 2024. 2

[70] Yuxue Yang, Lue Fan, Zuzeng Lin, Feng Wang, and Zhaoxiang Zhang. Layeranimate: Layer-level control for animation. In *ICCV*, pages 10865–10874, 2025. 5

[71] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2

[72] Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. In *ICLR*, 2024. 2, 6

[73] Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, et al. gsplat: An open-source library for gaussian splatting. *Journal of Machine Learning Research*, 26(34):1–17, 2025. 6

[74] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, pages 12–22, 2023. 4, 6, 1, 2

[75] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 2

[76] Mark YU, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *ICCV*, 2025. 2, 3, 6, 7

[77] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. 2, 4

[78] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024. 2, 6

[79] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 5

[80] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 5

[81] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *CVPR*, pages 21936–21947, 2025. 6

[82] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *ICCV*, pages 19855–19865, 2023. 5, 6, 1

NeoVerse: Enhancing 4D World Model with in-the-wild Monocular Videos

Supplementary Material

We provide videos on the project page[1] to vividly present qualitative results for an enhanced view experience.

A. Implementation Details

Reconstruction model. The transformer decoders in the bidirectional motion-encoding branch follow the design of DUSt3R [61], where each decoder block consists of a self-attention layer for intra-frame spatial modeling and a cross-attention layer for inter-frame temporal modeling. Finally, two DPT [49] heads are employed to predict the forward and backward motions, respectively. Here, we define the forward/backward velocities $\{\boldsymbol{v}_i^+, \boldsymbol{v}_i^-\}$ as the 3D displacements from the current frame to the next/previous frame in the camera coordinate.

Generation model. The multiple encoders for multi-modal conditions are implemented with 1) VAE [56] encoder for RGB images and depth maps, 2) convolutional layers with $8\times$ spatial and $4\times$ temporal compression ratio for masks and plüker embeddings. During the generation training stage, only convolutional layers are trainable while the VAE encoder is frozen.

B. Training Details

To ensure compatibility with the patch size of DINOv2 [47] in the reconstruction model ($\times 14$ downsampling) and the VAE in the generation model ($\times 8$ compression), we resize all input videos to have a longest edge of 560 pixels during reconstruction training, and a fixed resolution of 336×560 during generation training.

Reconstruction model. We train the reconstruction model on a combination of static and dynamic 3D datasets. For each training iteration, we sample N key frames (where $2 \leq N \leq 8$) and $N-1$ intermediate target frames between adjacent key frames. While only the N key frames are processed by the reconstruction model to predict Gaussians, the supervision loss is computed on all $2N-1$ frames. We utilize a cosine learning rate schedule with a peak learning rate of 1×10^{-4} and a warmup 5K iterations. To enhance the model's robustness to temporal direction, we apply a random temporal reversal augmentation with a probability of 0.5. The weights for the multi-task loss (Eq. 6 in the main paper) are set as follows: $\lambda_1 = 5.0$ (camera), $\lambda_2 = 1.0$ (depth), $\lambda_3 = 1.0$ (motion), and $\lambda_4 = 0.1$ (regularization).

[1] https://neoverse-4d.github.io

	Dataset	Dynamic	Depth	Pose	Flow	Real	Clip
①	PointOdyssey [82]	✓	✓	✓	✓		131
	DynamicReplica [31]	✓	✓	✓	✓		483
	Kubric [18]	✓	✓	✓	✓		5.7K
	Spring [46]	✓	✓	✓	✓		37
	VKITTI2 [7]	✓	✓	✓	✓		50
	Waymo [54]	✓	✓	✓	✓	✓	798
②	TartanAir [62]	✓	✓	✓			369
	BEDLAM [5]	✓	✓	✓			10.4K
	MVS-Synth [26]	✓	✓	✓			120
	GFIE [24]	✓	✓	✓		✓	81
③	HOI4D [41]	✓	✓			✓	3.0K
	CoP3D	✓		✓		✓	2.8K
④	DL3DV [38]		✓	✓	✓	✓	6.4K
	Scannet++ [74]		✓	✓	✓	✓	853
	ARKitScenes [4]		✓	✓	✓	✓	4.5K
	HyperSim [4]		✓	✓	✓	✓	457
	MapFree [1]		✓	✓	✓	✓	460
⑤	SpatialVID† [58]	✓	✓	✓		✓	371.3K
	Monocular Videos	✓				✓	**1M**

Table S1. **Training Datasets.** We categorize existing datasets into 5 groups based on their data characteristics. Group ①∼④ are used in reconstruction training, while group ⑤ is used in generation training. †: we only use videos for generation training.

Generation model. For the generation model, we use a constant learning rate of 1×10^{-5} and a batch size of 1 per GPU. To enable efficient on-the-fly reconstruction, we randomly sample $11 \sim 21$ keyframes from each video clip to reconstruct the 4DGS representation. Additionally, we employ a mask drop strategy where we randomly set all masks to 0 (indicating all degraded renderings need inpainting) with a probability of 0.2 to improve model robustness.

C. Dataset Details

We summarize the datasets used in our training in Table S1. Our training data is categorized into five groups:

① Dynamic datasets with 3D flow for velocity supervision.
② Dynamic datasets with depth and camera poses.
③ Dynamic datasets with incomplete 3D information (e.g., only camera poses or depth).
④ Static datasets (we assume 3D flow is zero).
⑤ Monocular videos.

We train the reconstruction model on ① to ④, while the generation model is trained on ⑤. Though SpatialVID provides 3D information, we don't use it for reconstruction training due to its unstable depth quality.

D. Evaluation Protocol

Following AnySplat, we perform test-time pose alignment to facilitate fair comparison, without introducing ground-truth poses during inference.

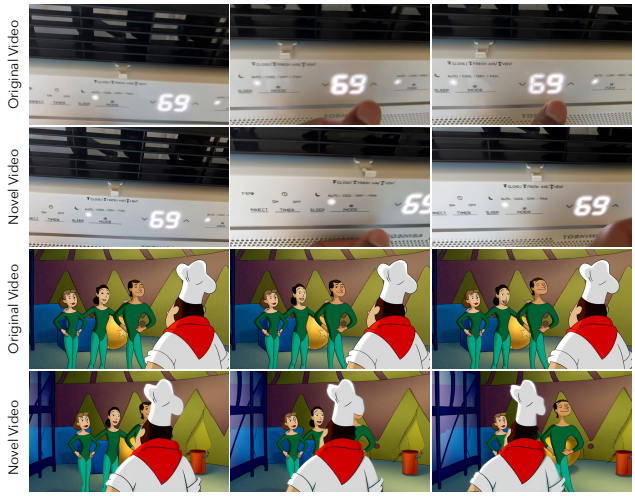


Figure S1. **Failure cases.** Top: Text generation failure. Bottom: Novel view generation on 2D data.

Static reconstruction. We evaluate static reconstruction performance on VRNeRF [66] and Scannet++ [74].

- **VRNeRF:** We select 6 scenes captured with pinhole cameras. For each scene, we randomly sample 16 views as input for reconstruction and 8 novel views for testing.
- **Scannet++:** We evaluate on all 50 scenes in the test set. We utilize 32 input views for reconstruction and evaluate on 16 novel views.

Dynamic reconstruction. For dynamic reconstruction on ADT [48], we follow 4DGT [67] to evaluate the same 4 scenes:

- Apartment_release_multiuser_cook_seq141_M1292
- Apartment_release_multiskeleton_party_seq114_M1292
- Apartment_release_meal_skeleton_seq135_M1292
- Apartment_release_work_skeleton_seq137_M1292

For each sequence, we sample a clip of 64 consecutive frames. We use 32 frames (stride 2) as input and the remaining 32 interleaved frames for testing.

For DyCheck [17], we evaluate 5 scenes (apple, block, paper-windmill, spin, teddy). We sample 64 consecutive timestamps for each scene, using 32 frames (stride 2) from a casually-captured video (camera 0) for reconstruction and the complete 64 frames from another fixed-camera video (camera 1) for testing.

E. Limitations and Failure Cases

Although our method can handle various challenging scenarios, there are some limitations as shown in Fig. S1. Similar to many video diffusion models, our method occasionally ***struggles to render legible and correct text*** (Top two rows). Besides, our method relies on extracting 3D clues from videos. It ***struggles with data lacking 3D geometry, such as 2D cartoons***. For instance, as the camera moves to the right side of a 2D cartoon character (Bottom two rows), the model may fail to generate the correct 3D profile (e.g., revealing the other side of a face), as the input video lacks inherent 3D structure.

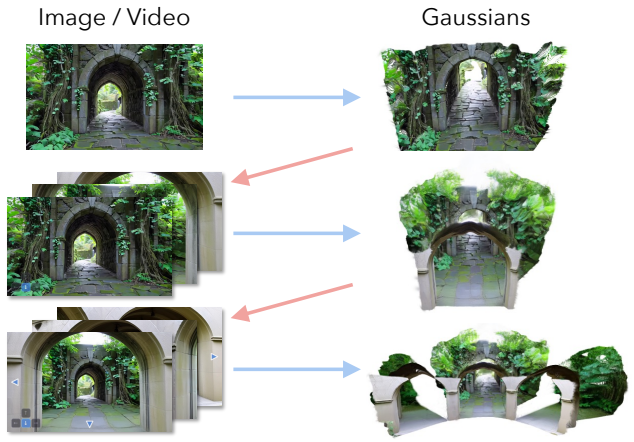


Figure S2. **Image to world.** Starting from a single view, NeoVerse can reconstruct a 3D scene, generate an exploration video, and iteratively expand the visible area.

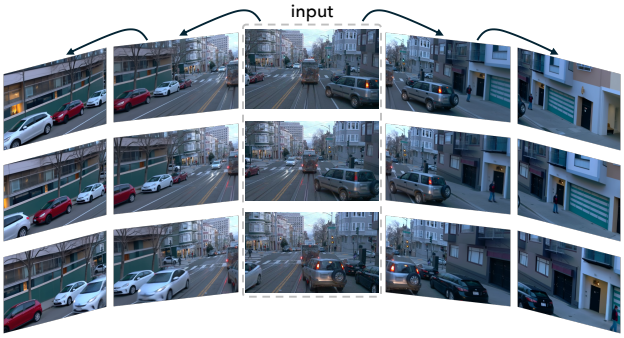


Figure S3. **Single-view to multi-view generation.** Starting from a single front-view video, NeoVerse can generate multi-view consistent videos.

F. Additional Qualitative Results

Image to world. Our NeoVerse allows for exploration in a captured image by iteratively generating new views and reconstructing the scene. As illustrated in Fig. S2, given a single starting image, we can generate a spatially coherent video trajectory. This generated video is then used to reconstruct a larger Gaussian Splatting scene, effectively "outpainting" the 3D world.

Single-view to multi-view Fig. S3 demonstrates the capability of generating multi-view consistent videos from a single-view video through **iterative application of NeoVerse**.